

Arduino Uno pinout –

Fuente de alimentación Hay 3 formas de alimentar el Arduino Uno:

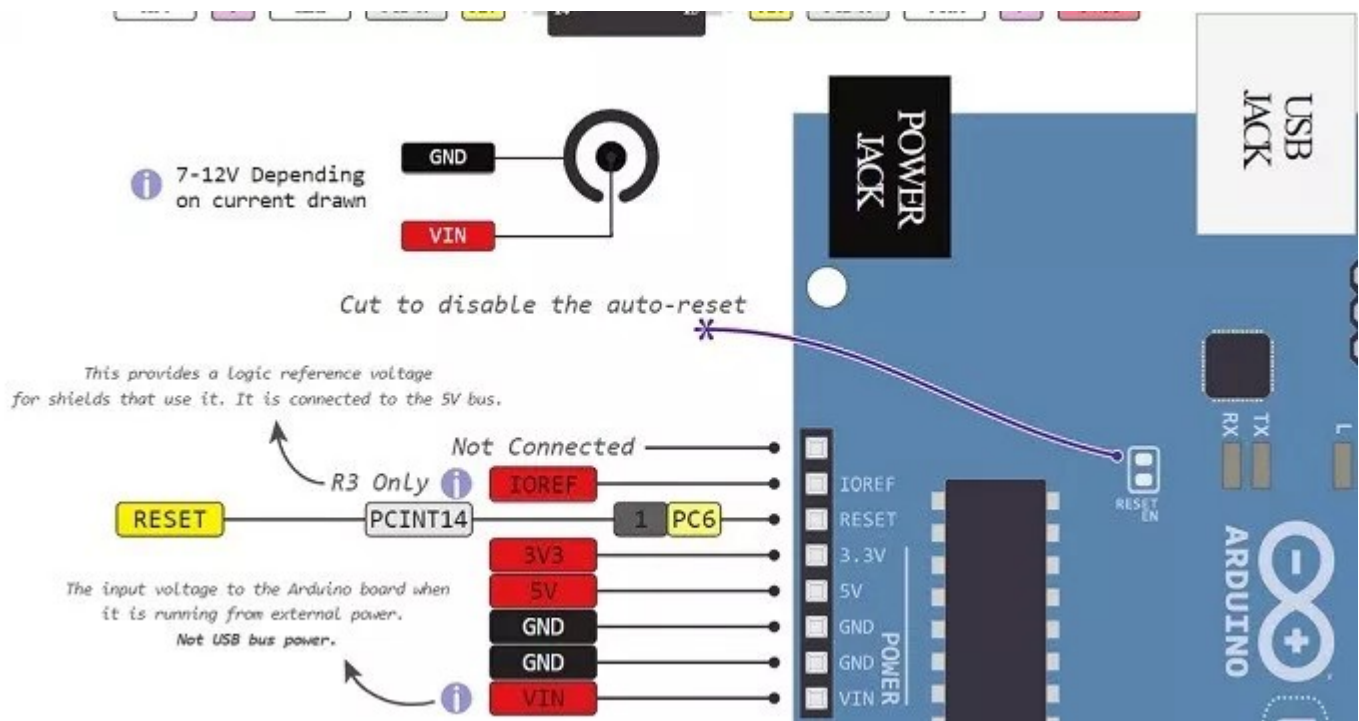
Barrel Jack –

El Barrel Jack, o DC Power Jack puede ser usado para alimentar tu placa Arduino. El conector cilíndrico suele estar conectado a un adaptador de pared.

La tarjeta puede ser alimentada por 5-20 voltios, pero el fabricante recomienda mantenerla entre 7-12 voltios. Por encima de 12 voltios, los reguladores podrían sobrecalentarse, y por debajo de 7 voltios, podrían no ser suficientes.

VIN Pin –

Este pin se utiliza para alimentar la placa Arduino Uno utilizando una fuente de alimentación externa. El voltaje debe estar dentro del rango mencionado anteriormente.



Cable USB –

Cuando se conecta a la computadora, proporciona 5 voltios a 500mA.

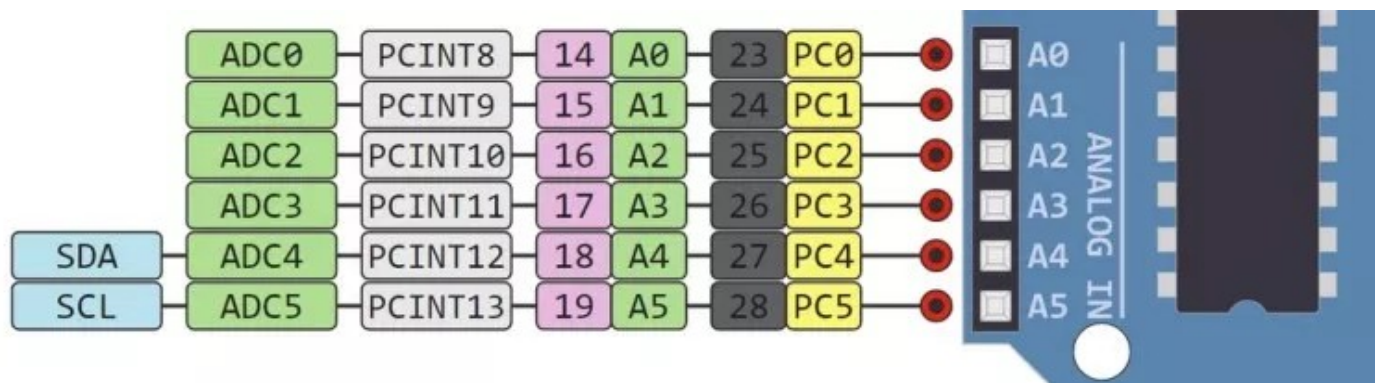
arduino Pinout alimentacion -

Arduino Uno Pinout, una sencilla introducción a su esquema Hay un diodo de protección de polaridad que se conecta entre el positivo de la clavija del barril y la clavija VIN, con una capacidad nominal de 1 amperio. La fuente de alimentación que utilices determina la potencia que tienes disponible para tu circuito. Por ejemplo, la alimentación del circuito usando el USB le limita a 500mA. Ten; en cuenta que también se utiliza para alimentar la MCU, los periféricos, los reguladores de a bordo y los componentes conectados a ella. Cuando alimentes tu circuito a través de la toma de barril o VIN, la capacidad máxima disponible está determinada por los reguladores de 5 y 3.3 voltios a bordo del Arduino.

5v y 3v3 Proporcionan 5 y 3.3v regulados para alimentar componentes externos de acuerdo con las especificaciones del fabricante.

GND En el pinout de Arduino Uno, puedes encontrar 5 pines GND, todos ellos interconectados. Las clavijas GND se utilizan para cerrar el circuito eléctrico y proporcionar un nivel de referencia lógico común en todo el circuito. Asegúrate siempre de que todos los GNDs (del Arduino, periféricos y componentes) estén conectados entre sí y tengan una conexión a tierra común.

RESET – reinicia el Arduino **IOREF** – Este pin es la referencia de entrada/salida. Proporciona la referencia de tensión con la que funciona el microcontrolador. Arduino Uno Pinout –



Analog IN La placa Arduino Uno tiene 6 pines analógicos, que utilizan ADC (Convertidor de Analógico a Digital). Estos pines sirven como entradas analógicas pero también pueden funcionar como entradas o salidas digitales. arduino Pinout pins Conversión de analógico a digital ADC son las siglas de Analog to Digital Converter. El ADC es un circuito electrónico utilizado para convertir señales analógicas en señales digitales. Esta representación digital de señales analógicas permite al procesador, que es un dispositivo digital, medir la señal analógica y utilizarla durante su funcionamiento.

Los pines Arduino A0-A5 son capaces de leer tensiones analógicas. En Arduino el ADC tiene una resolución de 10 bits, lo que significa que puede representar una tensión analógica de 1.024 niveles digitales. El ADC convierte el voltaje en bits que el microprocesador puede entender.

Un ejemplo común de un ADC es Voz sobre IP (VoIP). Cada smartphone tiene un micrófono que convierte las ondas sonoras (voz) en voltaje analógico. Este pasa a través del ADC del dispositivo, se convierte en datos digitales, que se transmiten a la parte receptora a través de Internet.

Arduino Uno Pinout –

Digital Pins Los pines 0-13 del Arduino Uno sirven como pines de entrada/salida digital.

El pin 13 del Arduino Uno está conectado al LED incorporado.

En el Arduino Uno – los pines 3,5,6,9,10,11 tienen capacidad PWM.

Es importante tener en cuenta: Cada clavija puede proporcionar hasta 40 mA máx. Pero la corriente recomendada es de 20 mA.

¿Qué significa digital?

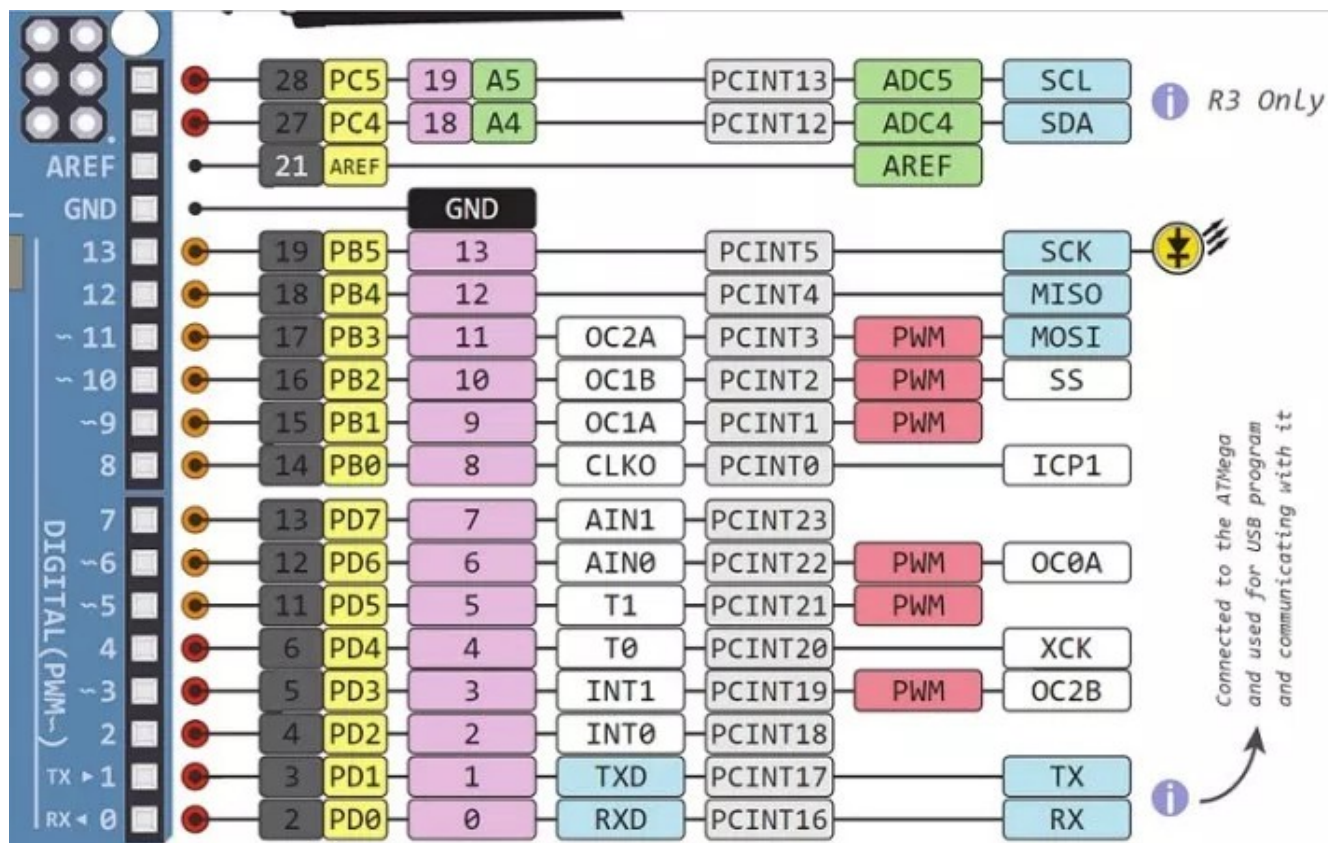
Digital es una forma de representar la tensión en 1 bit: 0 o 1. Los pines digitales del Arduino son pines diseñados para ser configurados como entradas o salidas según las necesidades del usuario. Los pines digitales están activados o desactivados. Cuando están en ON se encuentran en un estado de ALTA tensión de 5V y cuando están en OFF se encuentran en un estado de BAJA tensión de 0V. En el Arduino, cuando los pines digitales están configurados como salida, se ajustan a 0 o 5 voltios

Cuando los pines digitales se configuran como entrada, la tensión se suministra desde un dispositivo externo. Este voltaje puede variar entre 0-5 voltios, que se convierte en representación digital (0 o 1). Para determinar esto, hay dos umbrales:

Por debajo de 0.8v – considerado como 0.

Por encima de 2v – considerado como 1.

Cuando conectes un componente a un pin digital, asegúrate de que los niveles lógicos coincidan. Si la tensión se encuentra entre los umbrales, el valor de retorno será indefinido.



La corriente máxima absoluta proporcionada de todos los pines juntos es de 200 mA.

¿Qué es PWM?

En general, la modulación de ancho de pulso, PWM, es una técnica de modulación utilizada para codificar un mensaje en una señal pulsante. Un PWM se compone de dos componentes clave: frecuencia y ciclo de trabajo. La frecuencia PWM dicta el tiempo que se tarda en completar un solo ciclo (período) y la rapidez con la que la señal fluctúa de alta a baja. El ciclo de trabajo determina cuánto tiempo una señal permanece alta fuera del período total. El ciclo de trabajo se representa en porcentaje. En Arduino, los pines habilitados para PWM producen una frecuencia constante de ~ 500Hz, mientras que el ciclo de trabajo cambia de acuerdo a los parámetros establecidos por el usuario.

Protocolos de comunicación Serial (TTL) – Los pines digitales 0 y 1 son los pines seriales del Arduino Uno. Son utilizados por el módulo USB integrado.

¿Qué es la comunicación serie?

La comunicación serie o secuencial, se utiliza para intercambiar datos entre la placa Arduino y otro dispositivo en serie como ordenadores, pantallas, sensores y más. Cada placa Arduino tiene al menos un puerto serie. La comunicación serie se produce en los pines digitales 0 (RX) y 1 (TX), así como a través de USB. Arduino también soporta la comunicación serie a través de pines digitales con la librería SoftwareSerial Library. Esto permite al usuario conectar varios dispositivos habilitados para serie y dejar el puerto serie principal disponible para el USB.

Software serial y hardware serial –

La mayoría de los microcontroladores tienen hardware diseñado para comunicarse con otros dispositivos seriales. Los puertos serie de software utilizan un sistema de interrupción de cambio de pines para comunicarse. Hay una biblioteca incorporada para la comunicación serie de software. El software serial es utilizado por el procesador para simular puertos seriales adicionales. La única desventaja del software serial es que requiere más procesamiento y no puede soportar las mismas altas velocidades que el hardware serial.

Los pines SPI – SS/SCK/MISO/MOSI son los pines dedicados para la comunicación SPI. Se pueden encontrar en los pines digitales 10-13 del Arduino Uno y en las cabeceras del ICSP.

¿Qué es SPI?

La Interfaz Periférica Serial (SPI) es un protocolo de datos en serie utilizado por los microcontroladores para comunicarse con uno o más dispositivos externos en una conexión tipo bus. El SPI también se puede utilizar para conectar 2 microcontroladores. En el bus SPI, siempre hay un dispositivo que se denomina Maestro, Master, y todos los demás Esclavos, Slaves. En la mayoría de los casos, el microcontrolador es el dispositivo maestro. El pin SS (Slave Select) determina con qué dispositivo se está comunicando actualmente el Maestro.

Los dispositivos habilitados para SPI siempre tienen los siguientes pines:

MISO (Master In Slave Out) – Una línea para enviar datos al dispositivo Maestro.

MOSI (Master Out Slave In) – La línea Master para el envío de datos a dispositivos periféricos

SCK (Serial Clock) – Una señal de reloj generada por el dispositivo Master para sincronizar la transmisión de datos.

Los pines I2C – SCL/SDA son los pines dedicados para la comunicación I2C. En el Arduino Uno se encuentran en los pines analógicos A4 y A5.

¿Qué es I2C?

I2C es un protocolo de comunicación comúnmente conocido como “bus I2C”. El protocolo I2C fue diseñado para permitir la comunicación entre componentes en una sola tarjeta de circuito. Con I2C hay 2 cables denominados SCL y SDA.

SCL es la línea de reloj que está diseñada para sincronizar las transferencias de datos.

SDA es la línea utilizada para transmitir datos.

Cada dispositivo del bus I2C tiene una dirección única, se pueden conectar hasta 255 dispositivos en el mismo bus.

Aref – Tensión de referencia para las entradas analógicas.

Interrupción – **INT0 e INT1**. Arduino Uno tiene dos pines de interrupción externos.

Interrupción externa – Una interrupción externa es una interrupción del sistema que ocurre cuando hay interferencia externa. La interferencia puede provenir del usuario o de otros dispositivos de hardware de la red. Los usos comunes de estas interrupciones en Arduino son la lectura de la frecuencia de una onda cuadrada generada por los encoders o el despertar del procesador en un evento externo. Arduino tiene dos formas de interrupción:

Arduino Uno Pinout – Cabecera ICSP

ICSP son las siglas de In-Circuit Serial Programming. El nombre se originó a partir de las cabeceras de programación en el sistema (ISP). Fabricantes como Atmel que trabajan con Arduino han desarrollado sus propias cabeceras de programación en serie en circuito. Estos pines permiten al usuario programar el firmware de las placas Arduino. Hay seis pines ICSP disponibles en la placa Arduino que se pueden conectar a un dispositivo programador mediante un cable de programación.

